

## ソフトウェア工学(Software Engineering)

担当教員名	三浦 欽也		
学科・専攻、科目詳細	電気情報工学科 情報工学コース 5年 後期 1単位 講義		
学科のカリキュラム表	専門科目 必修科目		
共生システム工学の科目構成表	専門工学科目 専門応用系		
学習・教育目標	共生システム工学	F-1(55%)	G-2(25%) H-1(20%)
	JABEE基準1(1)	(d)(e)(h)	
科目的概要	ソフトウェアの需要が年々飛躍的に高まっているのに対して、ソフトウェアの生産性を急に改善するのは困難である。いわゆる「ソフトウェア危機」である。この「危機」を回避するためには、高品質のソフトウェアを効率的に開発する手法が求められている。この科目では、高品質のソフトウェアを効率的かつ組織的に開発するために発達してきたソフトウェア工学の、種々の技法・手法を講義する。		
テキスト(参考文献)	(教科書) 河村一樹:「改訂新版ソフトウェア工学入門」、近代科学社 (参考文献) Mint(経営情報研究会):「図解でわかるソフトウェア開発のすべて」、日本実業出版社		
履修上の注意	受講に当たっては、プログラミングII・III、データ構造とアルゴリズム、またはそれらに相当する科目を修得しておくことが望ましい。		
科目的達成目標	この科目では、一般的なウォーターフォールモデルによる段階的なソフトウェア開発のプロセスに沿って、各段階における目標と構造化技法について学習する(H-1, F-1)。その上で、近年急激に発展している、オブジェクト指向パラダイムに基づくソフトウェア開発技法についても学習する(G-2)。具体的な達成目標は以下の通りである。 [1] 一般的なソフトウェア開発工程を理解すること。 [2] 要求定義・プログラム設計・プログラミングの構造化技法を理解・習得すること。 [3] オブジェクト指向のソフトウェア開発技法について理解すること。これらのソフトウェア開発技法の学習を通じて、実践的なソフトウェア開発の能力を身につけ、実践的な問題解決能力を醸成する。		
自己学習	目標を達成するためには、授業以外に次の自己学習が必要である。 1) 適当な例を用いて、構造化分析を行ってみること。 2) 適当な例を用いて、構造化設計を行ってみること。 3) 適当な例を用いて、構造化チャートを記述してみること。 4) 適当な例を用いて、UMLのクラス図等を記述してみること。 5) ソフトウェア開発技法の最新の動向を調査し、概要を把握すること。		
目標達成度(成績)の評価方法と基準	合格の対象としない欠席条件(割合)		1/3以上の欠課
	評価方法: 後期中間試験(50%)、後期期末試験(50%) 評価基準: 達成目標の各々で修得すべき内容を以下に示す。 [1] ソフトウェアの一般的特性、ソフトウェア開発のプロセスモデル [2] 構造化分析、構造化設計、構造化プログラミング [3] オブジェクト指向パラダイム、オブジェクト指向分析・設計、UML 以上の内容を2回の定期試験(いずれも100点満点)で出題し、得点の平均が60点以上のものを合格とする。		
連絡先	miura@mail.kobe-c.ac.jp		

授業の計画・内容	
第1週	<b>ソフトウェアの本質</b> ソフトウェアの一般的定義・一般的構造・一般的特性について解説し、よいソフトウェアの条件を考察する。
第2週	<b>ソフトウェア工学の概要</b> ソフトウェア工学の発展経緯を概観し、その目標・意義・成果について簡単にまとめ、ソフトウェア工学の全体像を把握する。
第3週	<b>ソフトウェアのプロセスモデルと要求モデル</b> ソフトウェア開発工程のモデルであるプロセスモデルと、開発するべきソフトウェアのモデルである要求モデルについて解説し、それぞれいくつかのモデルを紹介する。
第4週	<b>構造化分析 1/2</b> ソフトウェアに求められるニーズを明確化して記述するための構造化技法の一つである、デマルコの構造化分析について解説する。
第5週	<b>構造化分析 2/2</b> 引き続きデマルコの構造化分析について解説し、構造化分析の問題点について触れ、構造化分析を実際に用いる際の手順と具体例を示す。
第6週	<b>構造化設計</b> プログラムのモジュール構造と、各モジュール間のインターフェースを設計するための構造化技法の一つである構造化設計について解説する。
第7週	<b>構造化プログラミング</b> プログラムモジュール内の制御構造を詳細に設計する際の構造化技法として、構造化プログラミングについて解説する。
第8週	<b>中間試験</b>
第9週	<b>オブジェクト指向の基礎概念 1/2</b> ソフトウェア開発のオブジェクト指向技法を学習するための基礎として、オブジェクト指向パラダイムの背景と基礎的な考え方を解説する。
第10週	<b>オブジェクト指向の基礎概念 2/2</b> 引き続きオブジェクト指向パラダイムの基本的な考え方を解説し、オブジェクトの構造やオブジェクト間の関係(関連・集約・汎化)等について解説する。
第11週	<b>オブジェクト指向プログラミング</b> オブジェクト指向パラダイムに基づくオブジェクト指向プログラミングについて概説する。
第12週	<b>オブジェクト指向分析・設計 1/2</b> オブジェクト指向技法による要求分析とプログラム設計を概説し、モデリングのための共通言語として近年重要性を増しているUMLについて概説する。
第13週	<b>オブジェクト指向分析・設計 2/2</b> 引き続きUMLについて概説し、オブジェクト指向技法によるソフトウェア開発の全体の流れについて概観する。
第14週	<b>ソフトウェアの品質管理とテスト技法</b> ソフトウェアの品質について考察し、その品質管理、および、ソフトウェアのテストの種類とテストケース設計の方法について概説する。
第15週	<b>種々のトピック</b> ソフトウェア開発に関連して、ソフトウェア開発環境・ソフトウェアの部品化/再利用・3階層アプリケーション等の概念について概説する。
<b>期末試験</b>	